Fig. 1

(Prior Art)

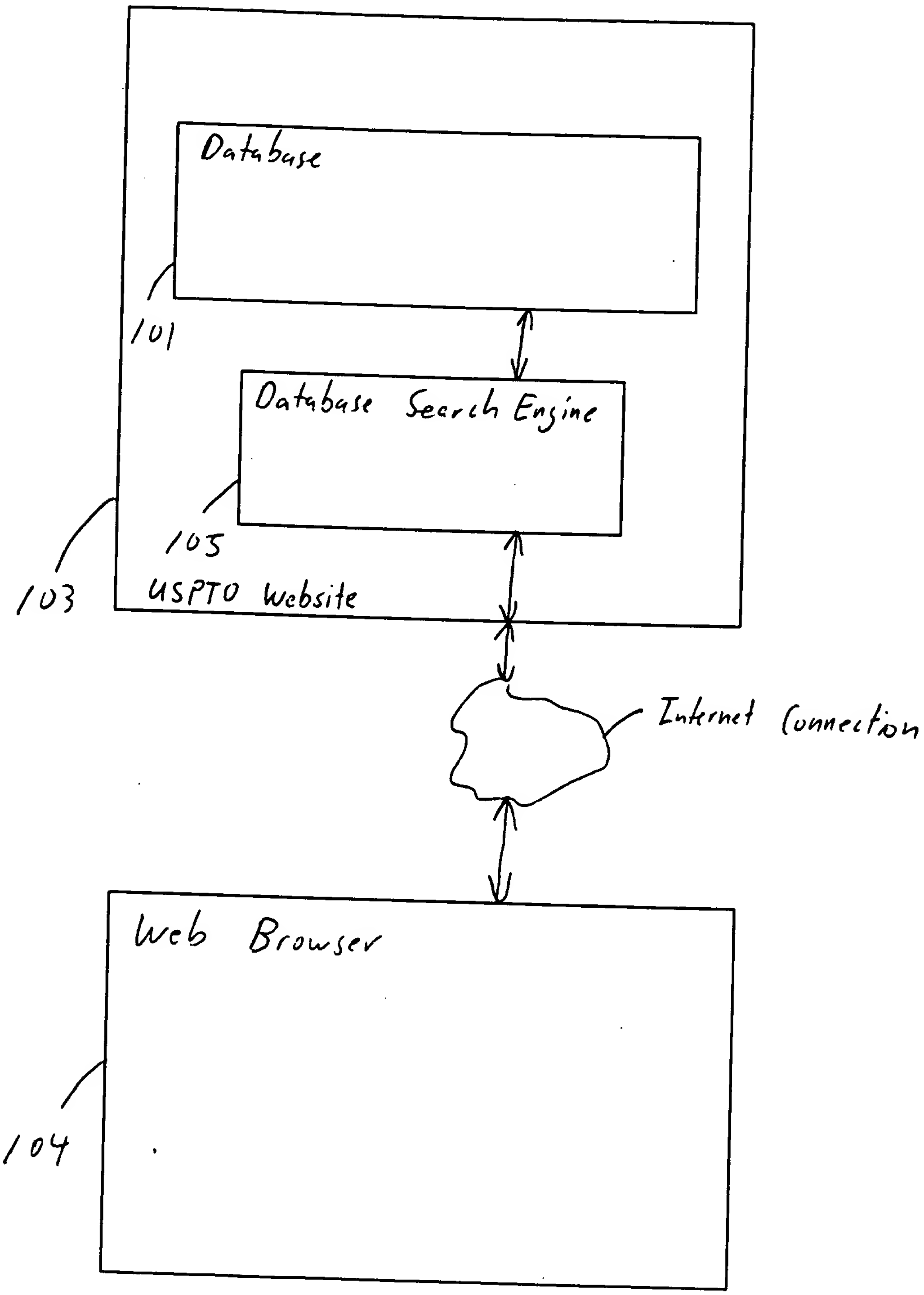


Fig. 2

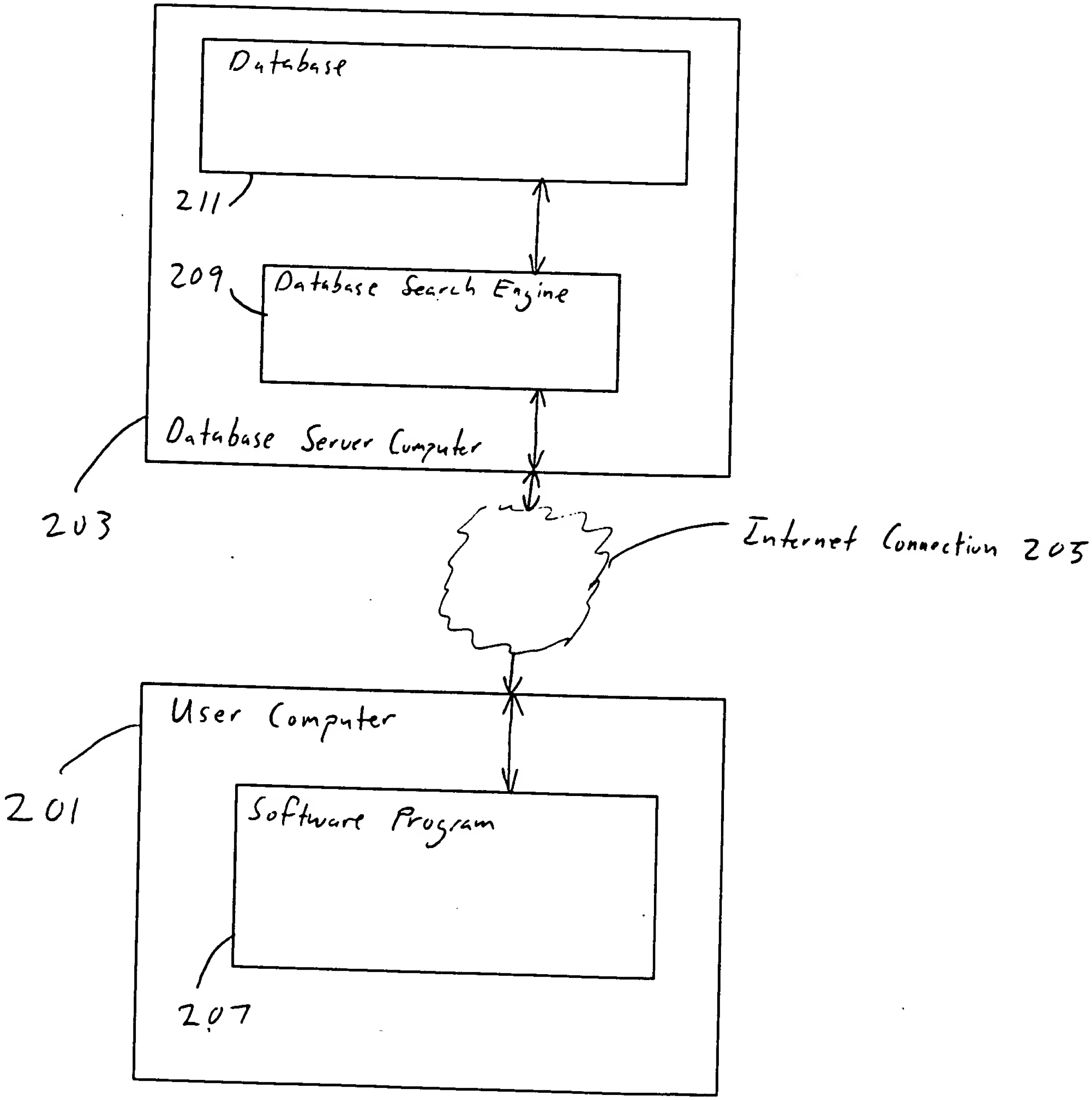
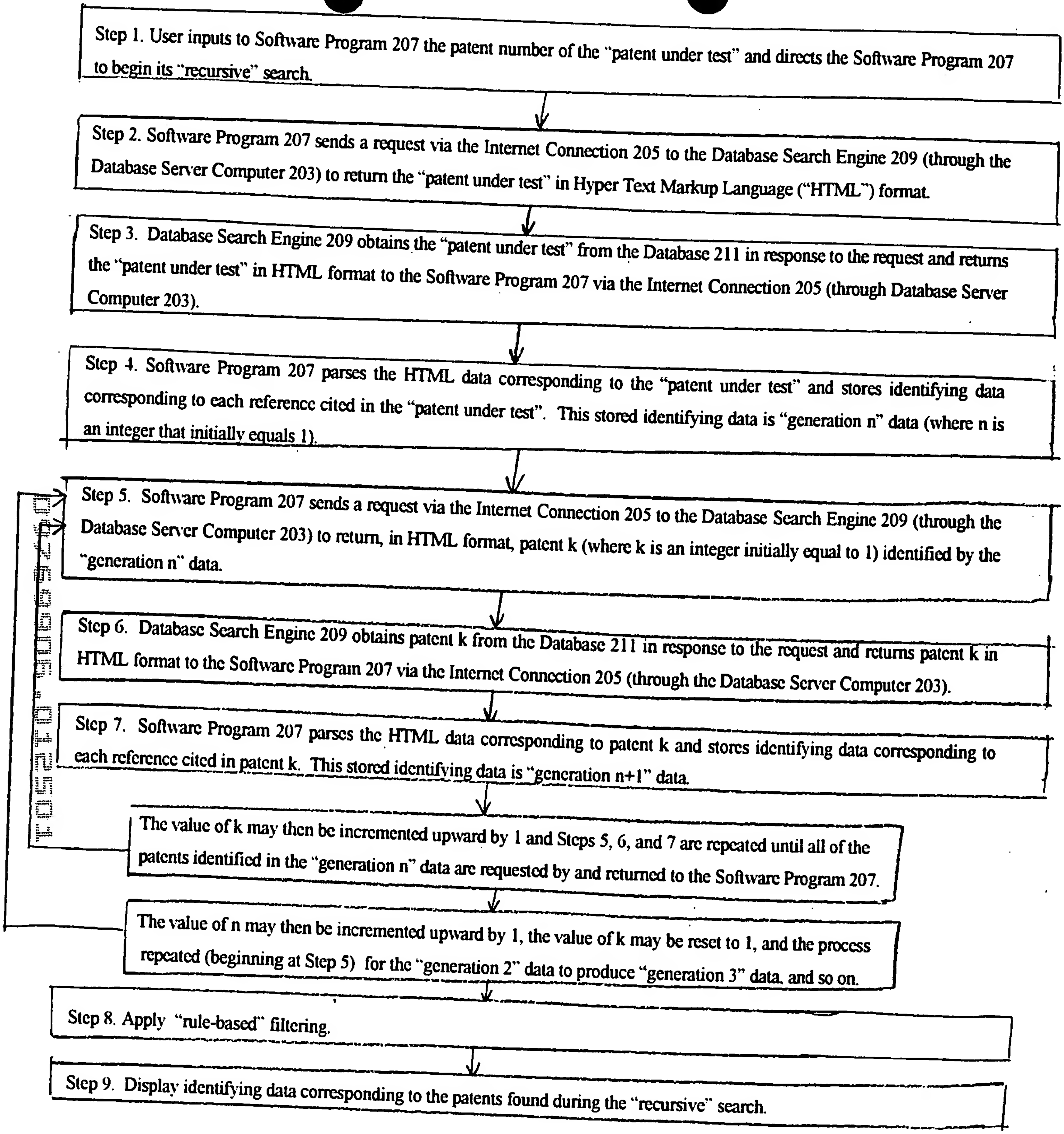

Database
211
209
Database Search Engine
Database Server Computer
203
Internet Connection 205
User Computer
201
Software Program
207

Fig. 3



Step 1. User inputs to Software Program 207 the patent number of the "patent under test" and directs the Software Program 207 to begin its "recursive" search.

Step 2. Software Program 207 sends a request via the Internet Connection 205 to the Database Search Engine 209 (through the Database Server Computer 203) to return the "patent under test" in Hyper Text Markup Language ("HTML") format.

Step 3. Database Search Engine 209 obtains the "patent under test" from the Database 211 in response to the request and returns the "patent under test" in HTML format to the Software Program 207 via the Internet Connection 205 (through Database Server Computer 203).

Step 4. Software Program 207 parses the HTML data corresponding to the "patent under test" and stores identifying data corresponding to each reference cited in the "patent under test". This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 5. Software Program 207 sends a request via the Internet Connection 205 to the Database Search Engine 209 (through the Database Server Computer 203) to return, in HTML format, patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 6. Database Search Engine 209 obtains patent k from the Database 211 in response to the request and returns patent k in HTML format to the Software Program 207 via the Internet Connection 205 (through the Database Server Computer 203).

Step 7. Software Program 207 parses the HTML data corresponding to patent k and stores identifying data corresponding to each reference cited in patent k. This stored identifying data is "generation n+1" data.

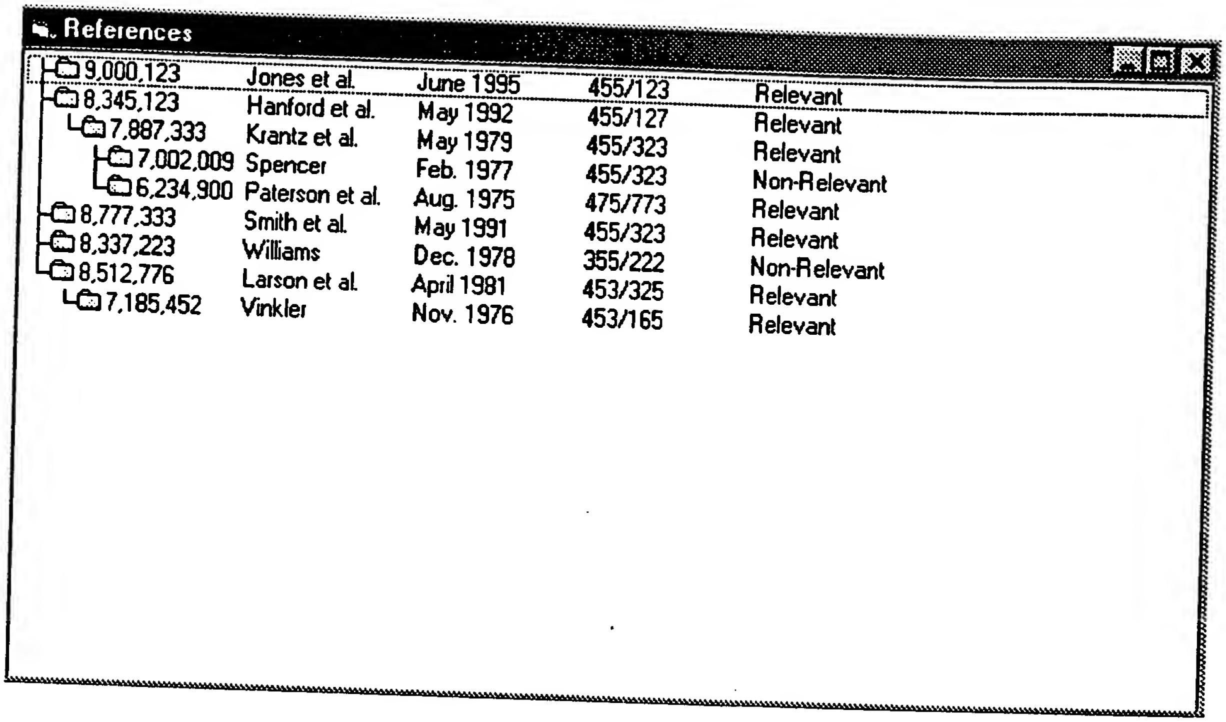
The value of k may then be incremented upward by 1 and Steps 5, 6, and 7 are repeated until all of the patents identified in the "generation n" data are requested by and returned to the Software Program 207.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 5) for the "generation 2" data to produce "generation 3" data, and so on.

Step 8. Apply "rule-based" filtering.

Step 9. Display identifying data corresponding to the patents found during the "recursive" search.

Fig. 4



References

```
├🗀 9,000,123      Jones et al.      June 1995   455/123   Relevant
├🗀 8,345,123      Hanford et al.    May 1992    455/127   Relevant
│ └🗀 7,887,333    Krantz et al.     May 1979    455/323   Relevant
│   ├🗀 7,002,009  Spencer           Feb. 1977   455/323   Non-Relevant
│   └🗀 6,234,900  Paterson et al.   Aug. 1975   475/773   Relevant
├🗀 8,777,333      Smith et al.      May 1991    455/323   Relevant
├🗀 8,337,223      Williams          Dec. 1978   355/222   Relevant
└🗀 8,512,776      Larson et al.     April 1981  453/325   Non-Relevant
  └🗀 7,185,452    Vinkler           Nov. 1976   453/165   Relevant
```

Fig. 5

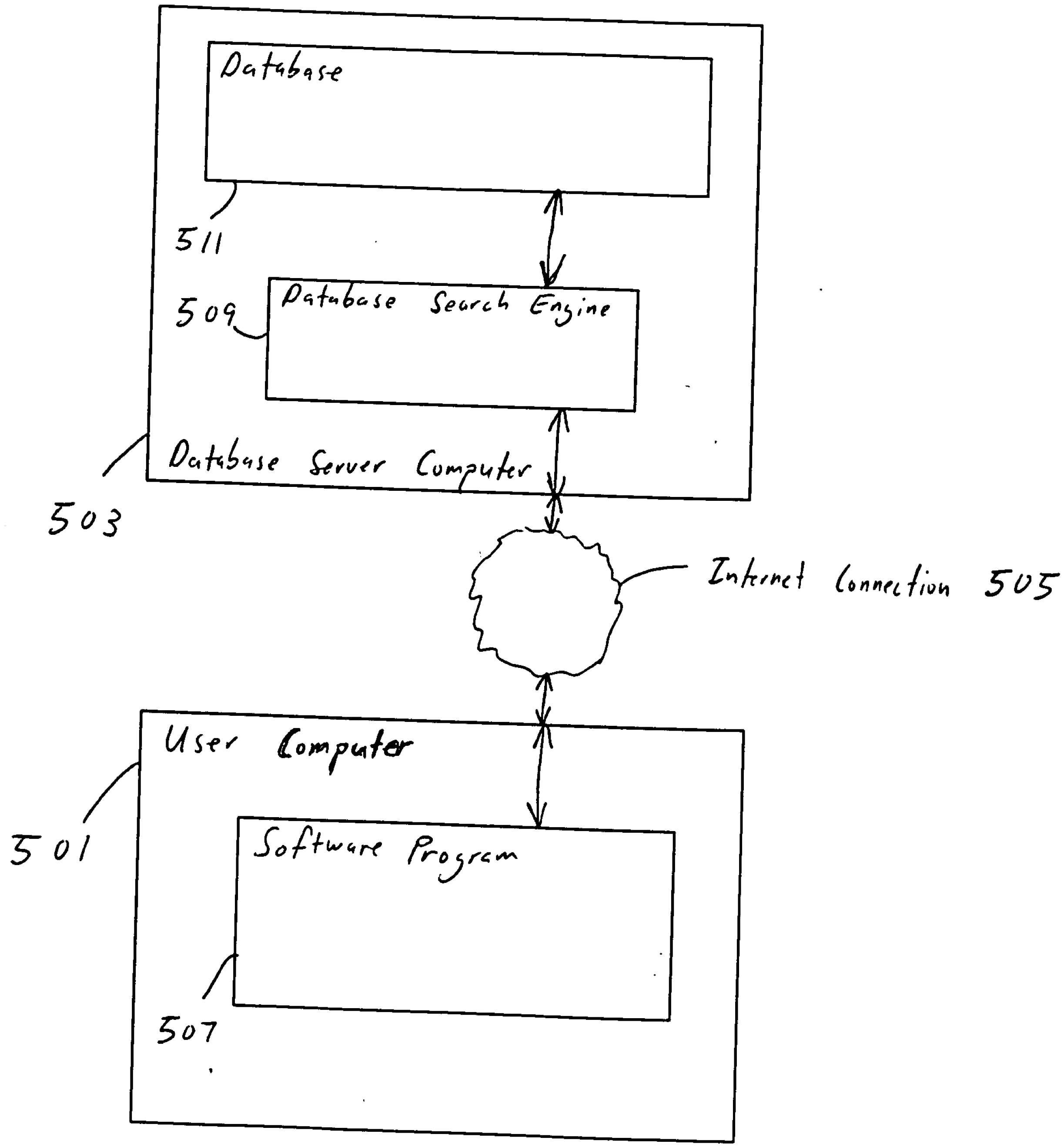
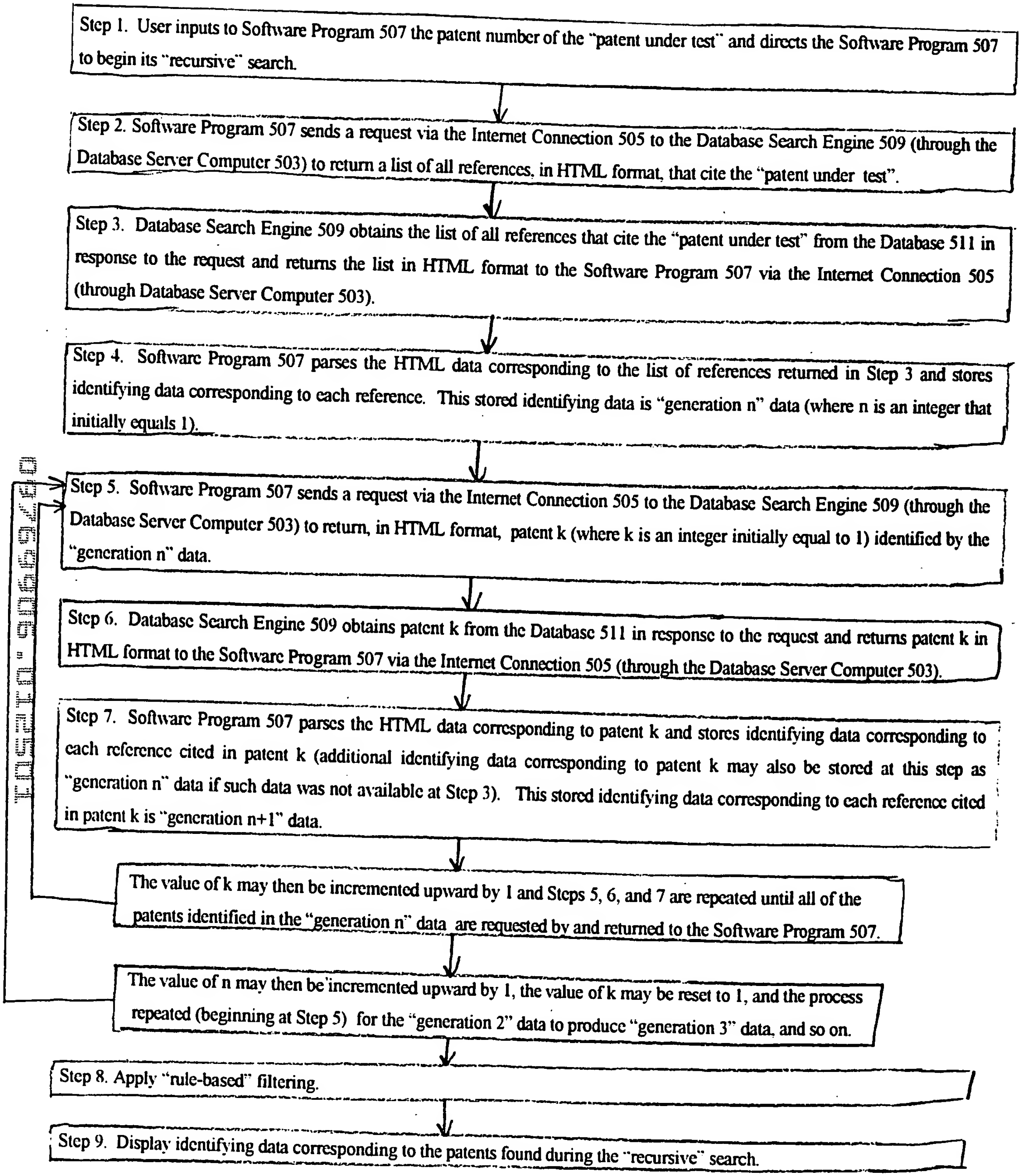

Database
511
509
Database Search Engine
Database Server Computer
503
Internet Connection 505
User Computer
501
Software Program
507

Fig. 6



Step 1. User inputs to Software Program 507 the patent number of the "patent under test" and directs the Software Program 507 to begin its "recursive" search.

Step 2. Software Program 507 sends a request via the Internet Connection 505 to the Database Search Engine 509 (through the Database Server Computer 503) to return a list of all references, in HTML format, that cite the "patent under test".

Step 3. Database Search Engine 509 obtains the list of all references that cite the "patent under test" from the Database 511 in response to the request and returns the list in HTML format to the Software Program 507 via the Internet Connection 505 (through Database Server Computer 503).

Step 4. Software Program 507 parses the HTML data corresponding to the list of references returned in Step 3 and stores identifying data corresponding to each reference. This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 5. Software Program 507 sends a request via the Internet Connection 505 to the Database Search Engine 509 (through the Database Server Computer 503) to return, in HTML format, patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 6. Database Search Engine 509 obtains patent k from the Database 511 in response to the request and returns patent k in HTML format to the Software Program 507 via the Internet Connection 505 (through the Database Server Computer 503).

Step 7. Software Program 507 parses the HTML data corresponding to patent k and stores identifying data corresponding to each reference cited in patent k (additional identifying data corresponding to patent k may also be stored at this step as "generation n" data if such data was not available at Step 3). This stored identifying data corresponding to each reference cited in patent k is "generation n+1" data.

The value of k may then be incremented upward by 1 and Steps 5, 6, and 7 are repeated until all of the patents identified in the "generation n" data are requested by and returned to the Software Program 507.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 5) for the "generation 2" data to produce "generation 3" data, and so on.

Step 8. Apply "rule-based" filtering.

Step 9. Display identifying data corresponding to the patents found during the "recursive" search.

Fig. 7

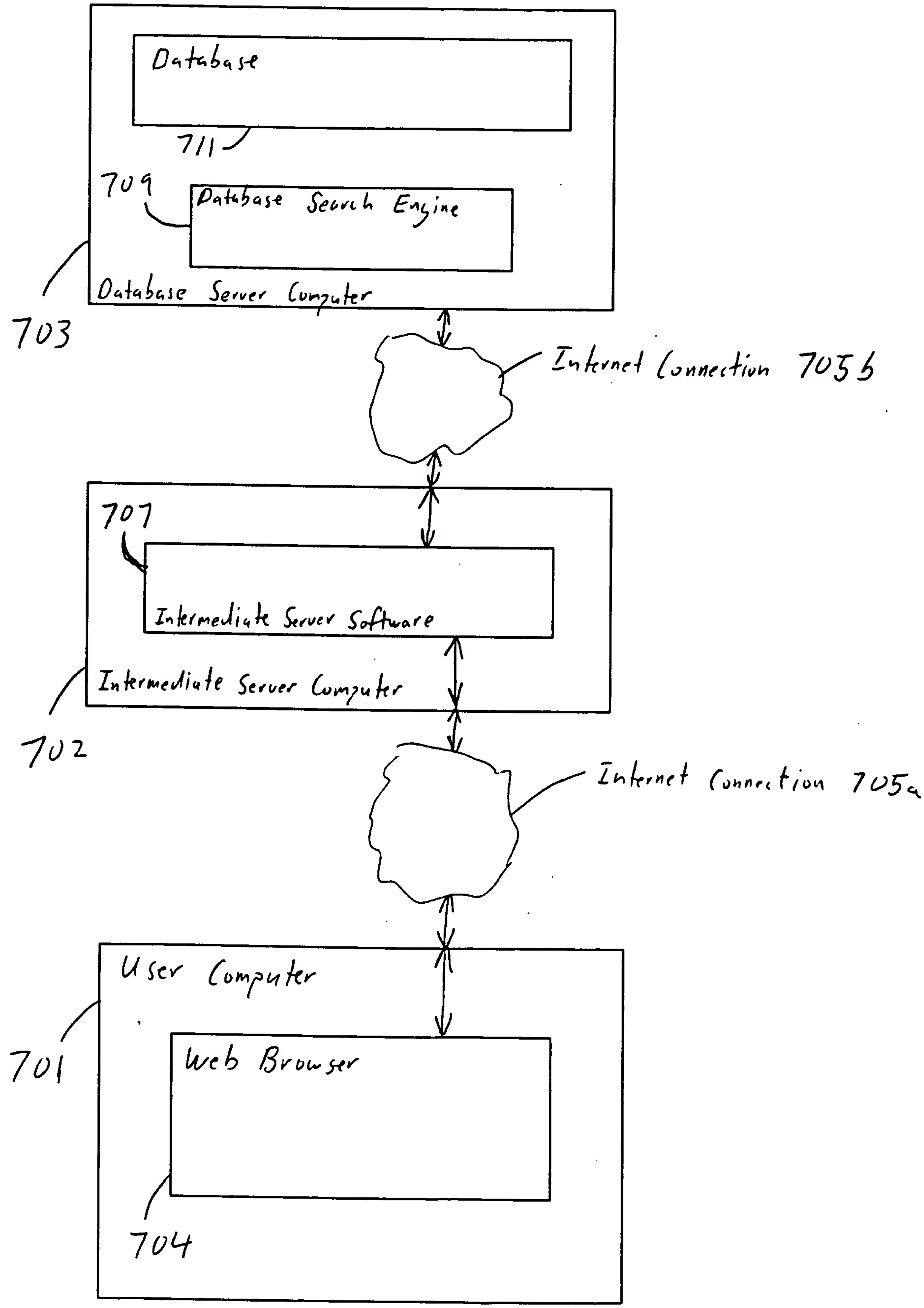
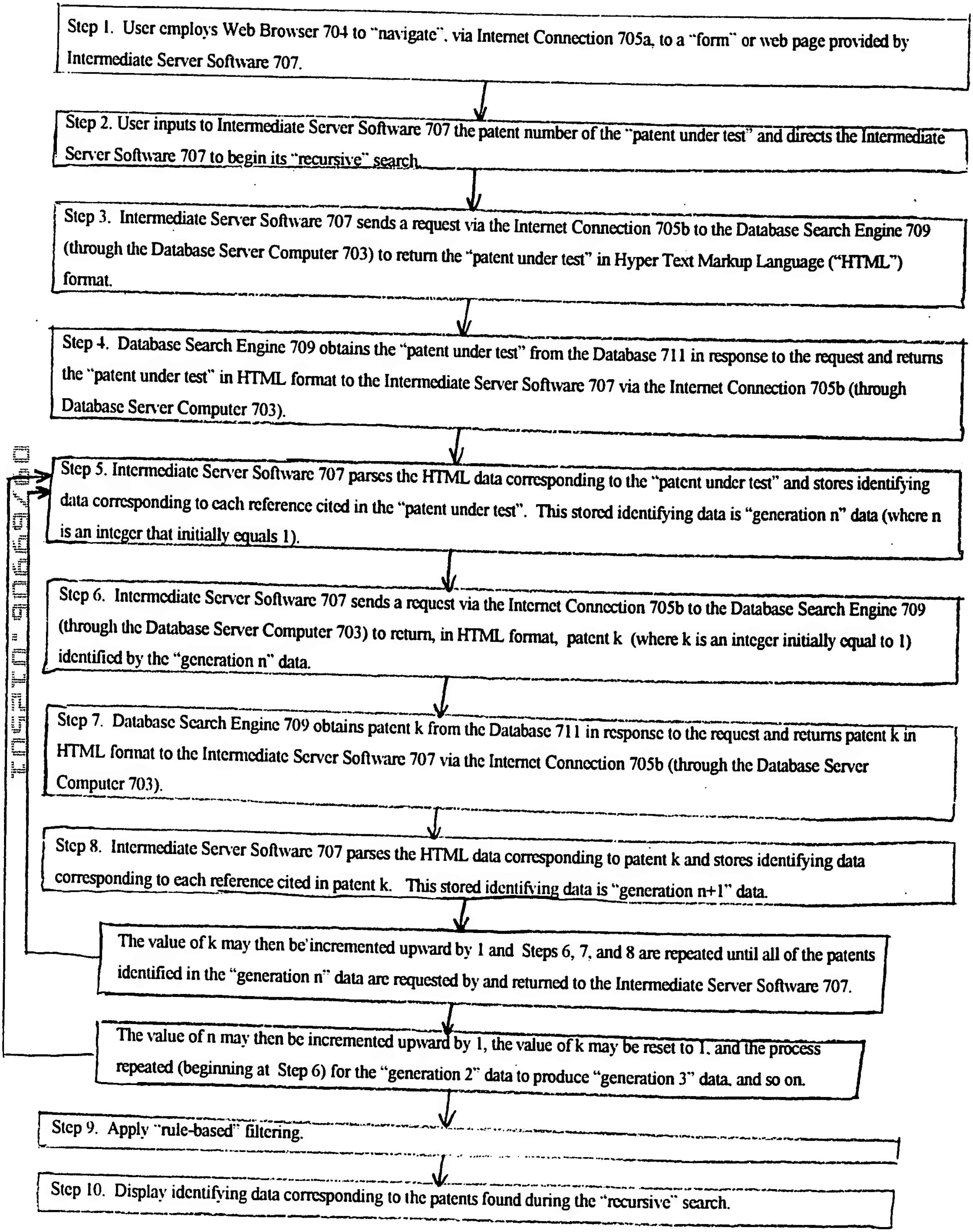

Database
711
709
Database Search Engine
Database Server Computer
703
Internet Connection 705b
707
Intermediate Server Software
Intermediate Server Computer
702
Internet Connection 705a
User Computer
701
Web Browser
704

Fig. 8



Step 1. User employs Web Browser 704 to "navigate", via Internet Connection 705a, to a "form" or web page provided by Intermediate Server Software 707.

Step 2. User inputs to Intermediate Server Software 707 the patent number of the "patent under test" and directs the Intermediate Server Software 707 to begin its "recursive" search.

Step 3. Intermediate Server Software 707 sends a request via the Internet Connection 705b to the Database Search Engine 709 (through the Database Server Computer 703) to return the "patent under test" in Hyper Text Markup Language ("HTML") format.

Step 4. Database Search Engine 709 obtains the "patent under test" from the Database 711 in response to the request and returns the "patent under test" in HTML format to the Intermediate Server Software 707 via the Internet Connection 705b (through Database Server Computer 703).

Step 5. Intermediate Server Software 707 parses the HTML data corresponding to the "patent under test" and stores identifying data corresponding to each reference cited in the "patent under test". This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 6. Intermediate Server Software 707 sends a request via the Internet Connection 705b to the Database Search Engine 709 (through the Database Server Computer 703) to return, in HTML format, patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 7. Database Search Engine 709 obtains patent k from the Database 711 in response to the request and returns patent k in HTML format to the Intermediate Server Software 707 via the Internet Connection 705b (through the Database Server Computer 703).

Step 8. Intermediate Server Software 707 parses the HTML data corresponding to patent k and stores identifying data corresponding to each reference cited in patent k. This stored identifying data is "generation n+1" data.

The value of k may then be incremented upward by 1 and Steps 6, 7, and 8 are repeated until all of the patents identified in the "generation n" data are requested by and returned to the Intermediate Server Software 707.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 6) for the "generation 2" data to produce "generation 3" data, and so on.

Step 9. Apply "rule-based" filtering.

Step 10. Display identifying data corresponding to the patents found during the "recursive" search.

Fig. 9

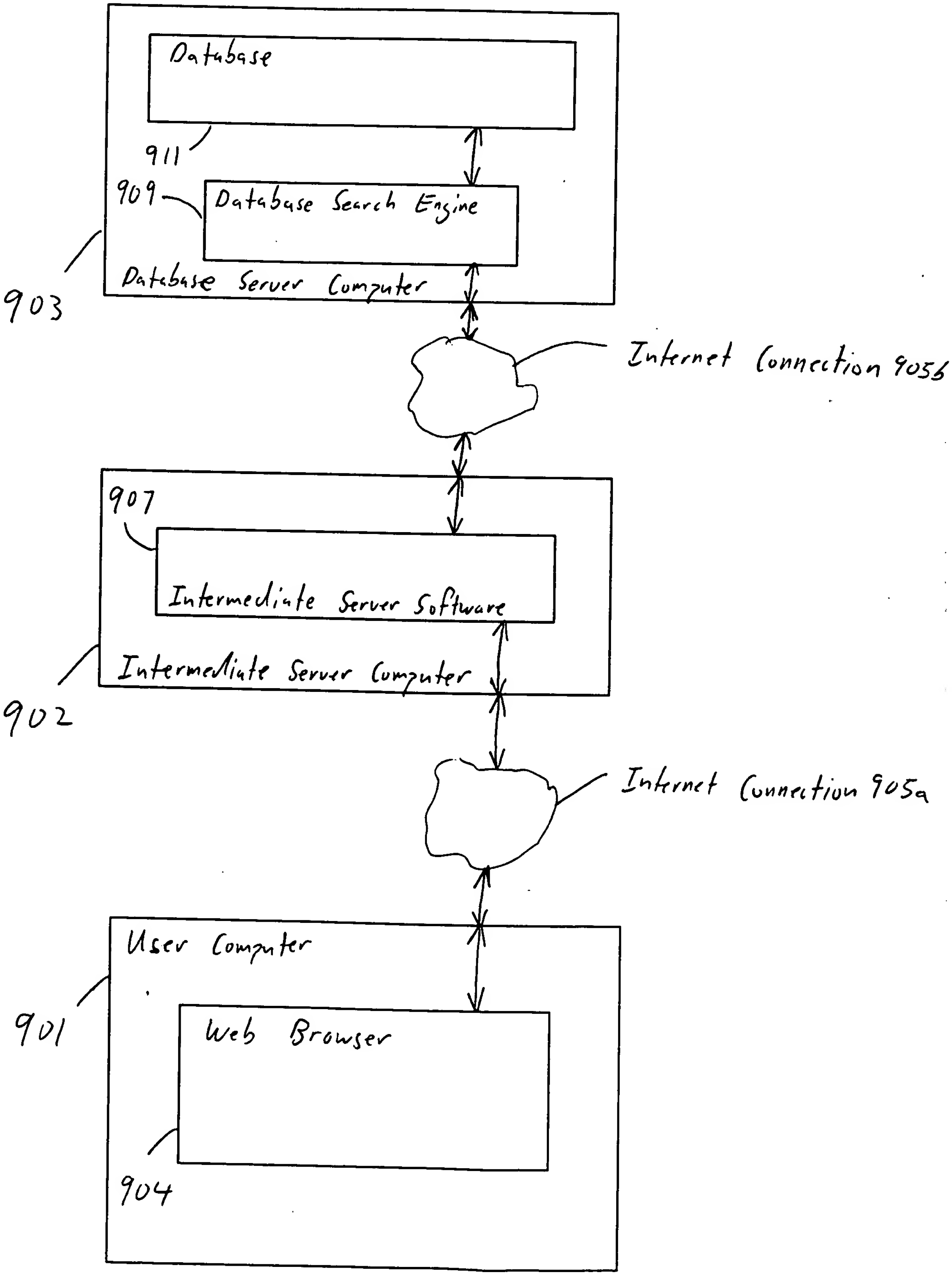
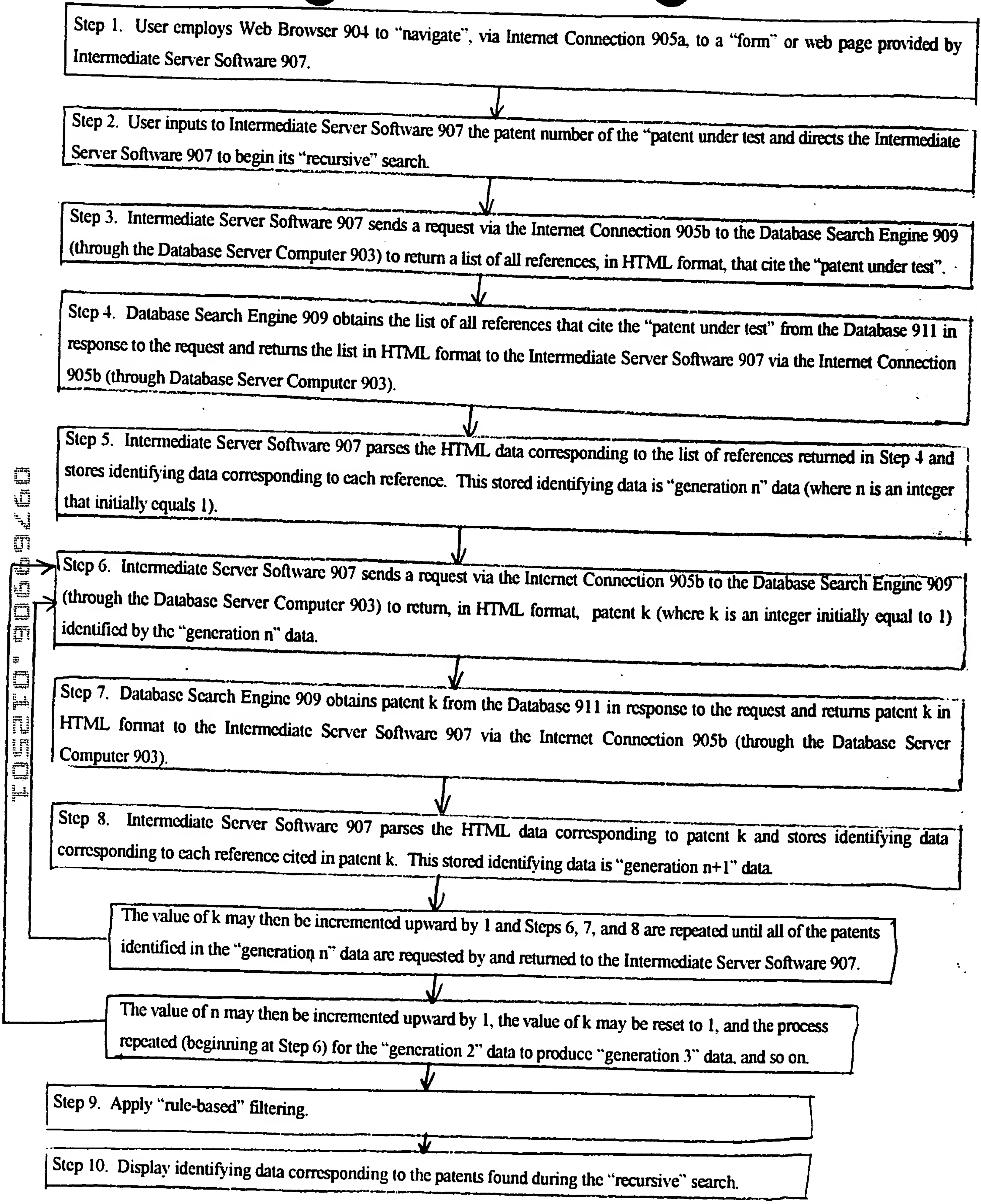

Database
911
909
Database Search Engine
Database Server Computer
903
Internet Connection 905b
907
Intermediate Server Software
Intermediate Server Computer
902
Internet Connection 905a
User Computer
901
Web Browser
904

Fig. 10



Step 1. User employs Web Browser 904 to "navigate", via Internet Connection 905a, to a "form" or web page provided by Intermediate Server Software 907.

Step 2. User inputs to Intermediate Server Software 907 the patent number of the "patent under test and directs the Intermediate Server Software 907 to begin its "recursive" search.

Step 3. Intermediate Server Software 907 sends a request via the Internet Connection 905b to the Database Search Engine 909 (through the Database Server Computer 903) to return a list of all references, in HTML format, that cite the "patent under test".

Step 4. Database Search Engine 909 obtains the list of all references that cite the "patent under test" from the Database 911 in response to the request and returns the list in HTML format to the Intermediate Server Software 907 via the Internet Connection 905b (through Database Server Computer 903).

Step 5. Intermediate Server Software 907 parses the HTML data corresponding to the list of references returned in Step 4 and stores identifying data corresponding to each reference. This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 6. Intermediate Server Software 907 sends a request via the Internet Connection 905b to the Database Search Engine 909 (through the Database Server Computer 903) to return, in HTML format, patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 7. Database Search Engine 909 obtains patent k from the Database 911 in response to the request and returns patent k in HTML format to the Intermediate Server Software 907 via the Internet Connection 905b (through the Database Server Computer 903).

Step 8. Intermediate Server Software 907 parses the HTML data corresponding to patent k and stores identifying data corresponding to each reference cited in patent k. This stored identifying data is "generation n+1" data.

The value of k may then be incremented upward by 1 and Steps 6, 7, and 8 are repeated until all of the patents identified in the "generation n" data are requested by and returned to the Intermediate Server Software 907.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 6) for the "generation 2" data to produce "generation 3" data, and so on.

Step 9. Apply "rule-based" filtering.

Step 10. Display identifying data corresponding to the patents found during the "recursive" search.

Fig. 11

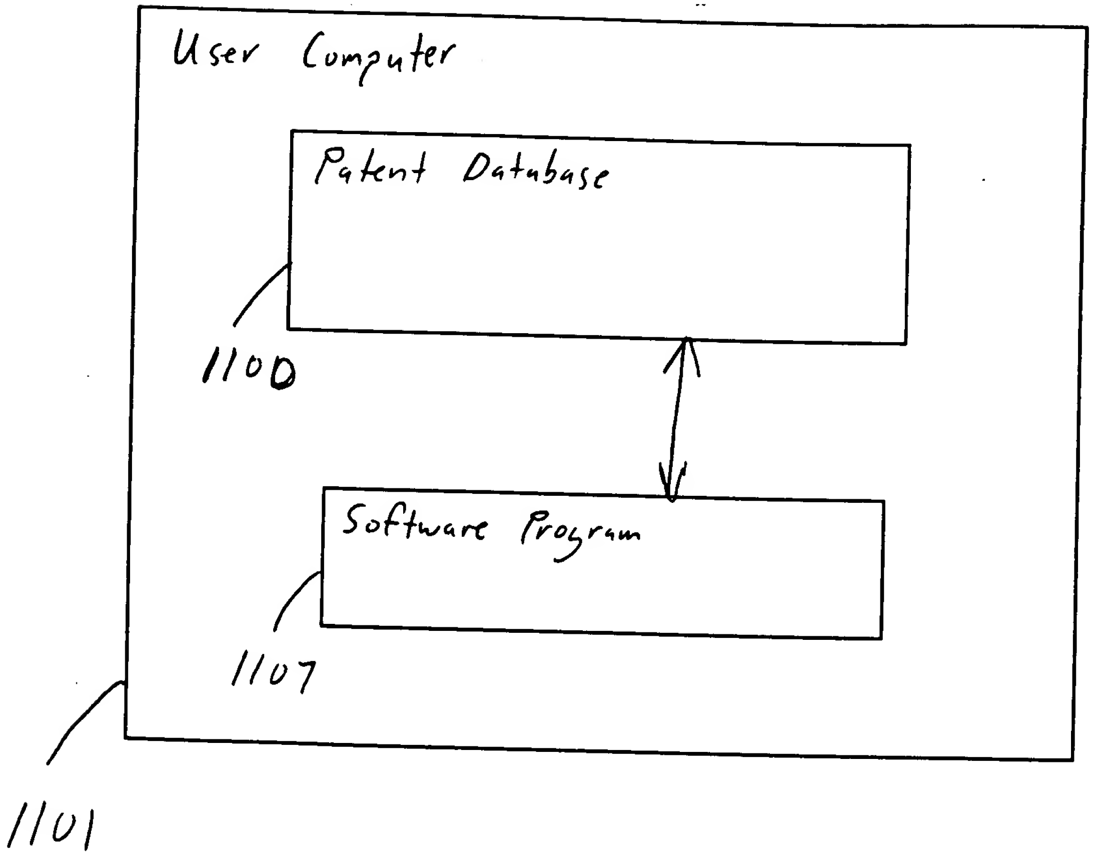
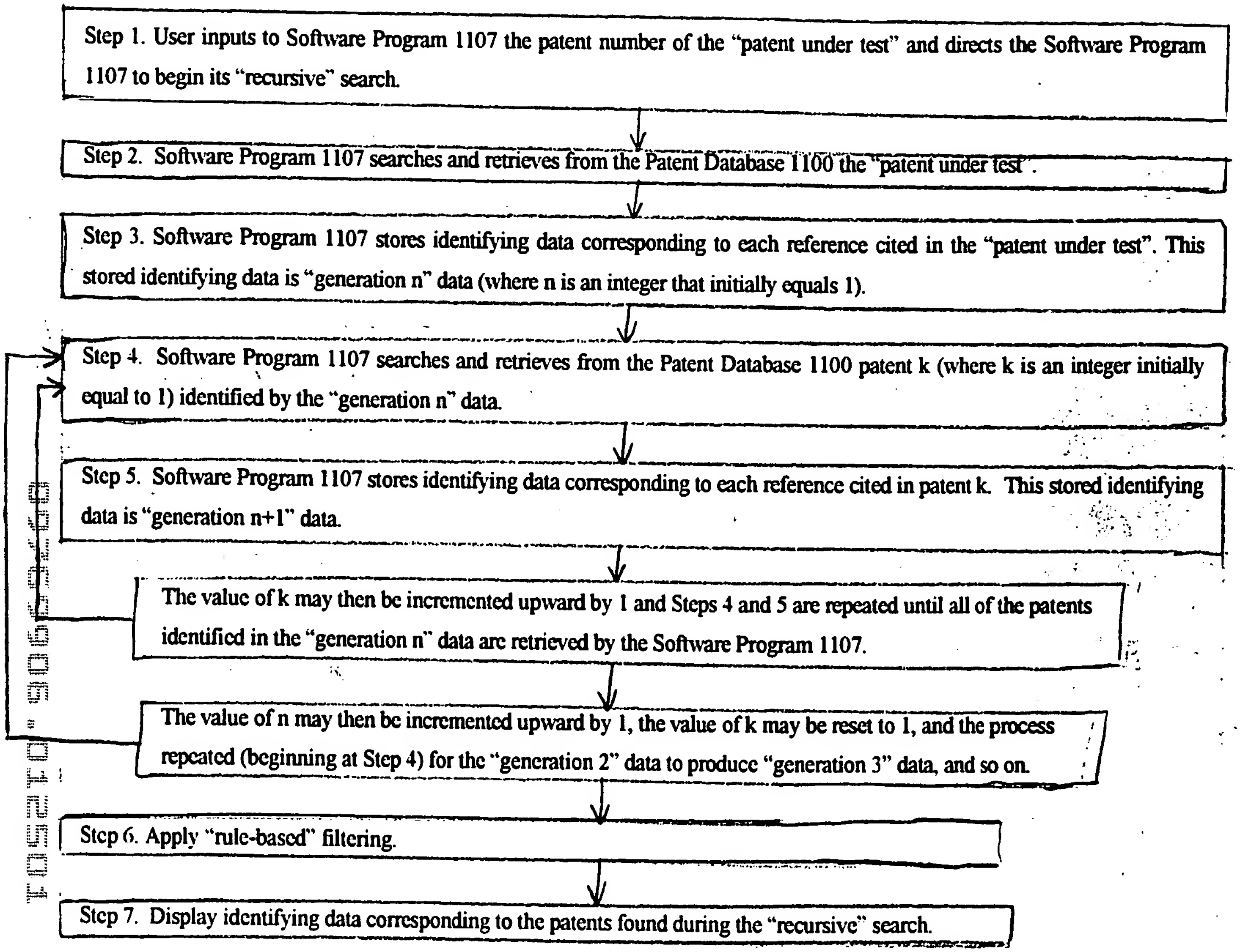

User Computer
Patent Database
1100
Software Program
1107
1101

Fig. 12



Step 1. User inputs to Software Program 1107 the patent number of the "patent under test" and directs the Software Program 1107 to begin its "recursive" search.

Step 2. Software Program 1107 searches and retrieves from the Patent Database 1100 the "patent under test".

Step 3. Software Program 1107 stores identifying data corresponding to each reference cited in the "patent under test". This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 4. Software Program 1107 searches and retrieves from the Patent Database 1100 patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 5. Software Program 1107 stores identifying data corresponding to each reference cited in patent k. This stored identifying data is "generation n+1" data.

The value of k may then be incremented upward by 1 and Steps 4 and 5 are repeated until all of the patents identified in the "generation n" data are retrieved by the Software Program 1107.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 4) for the "generation 2" data to produce "generation 3" data, and so on.

Step 6. Apply "rule-based" filtering.

Step 7. Display identifying data corresponding to the patents found during the "recursive" search.

Fig. 13

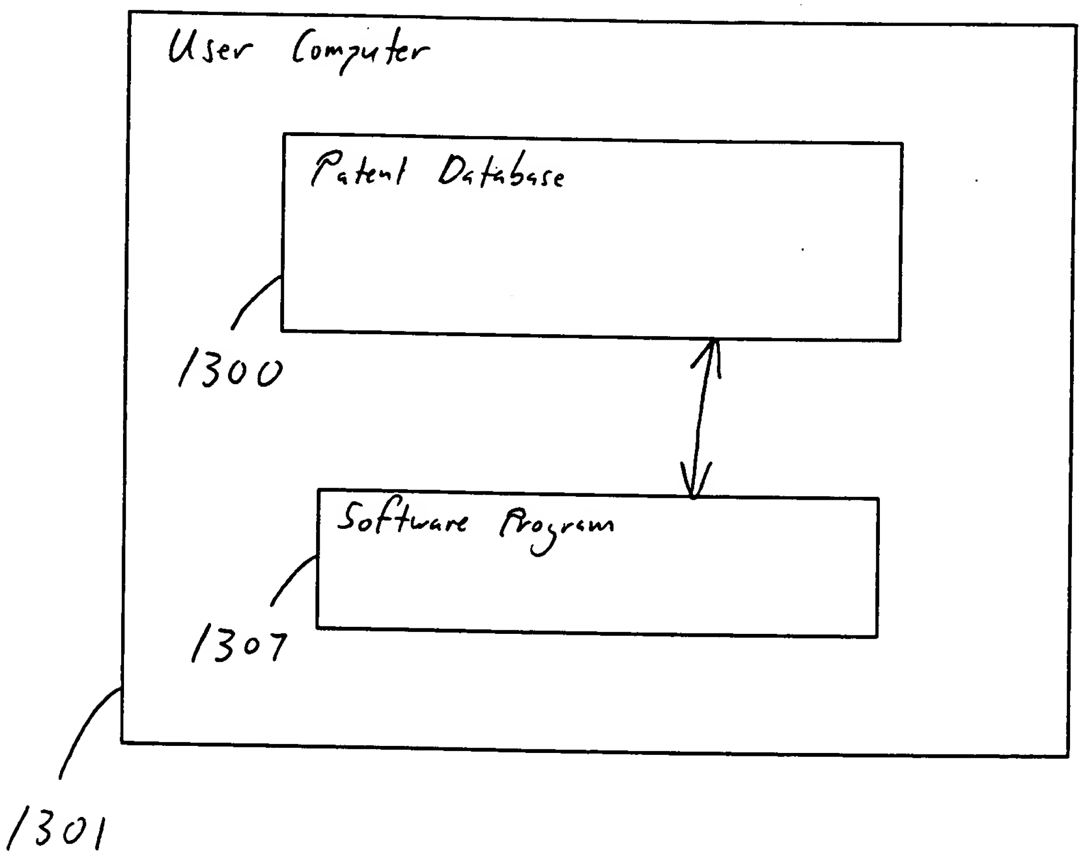
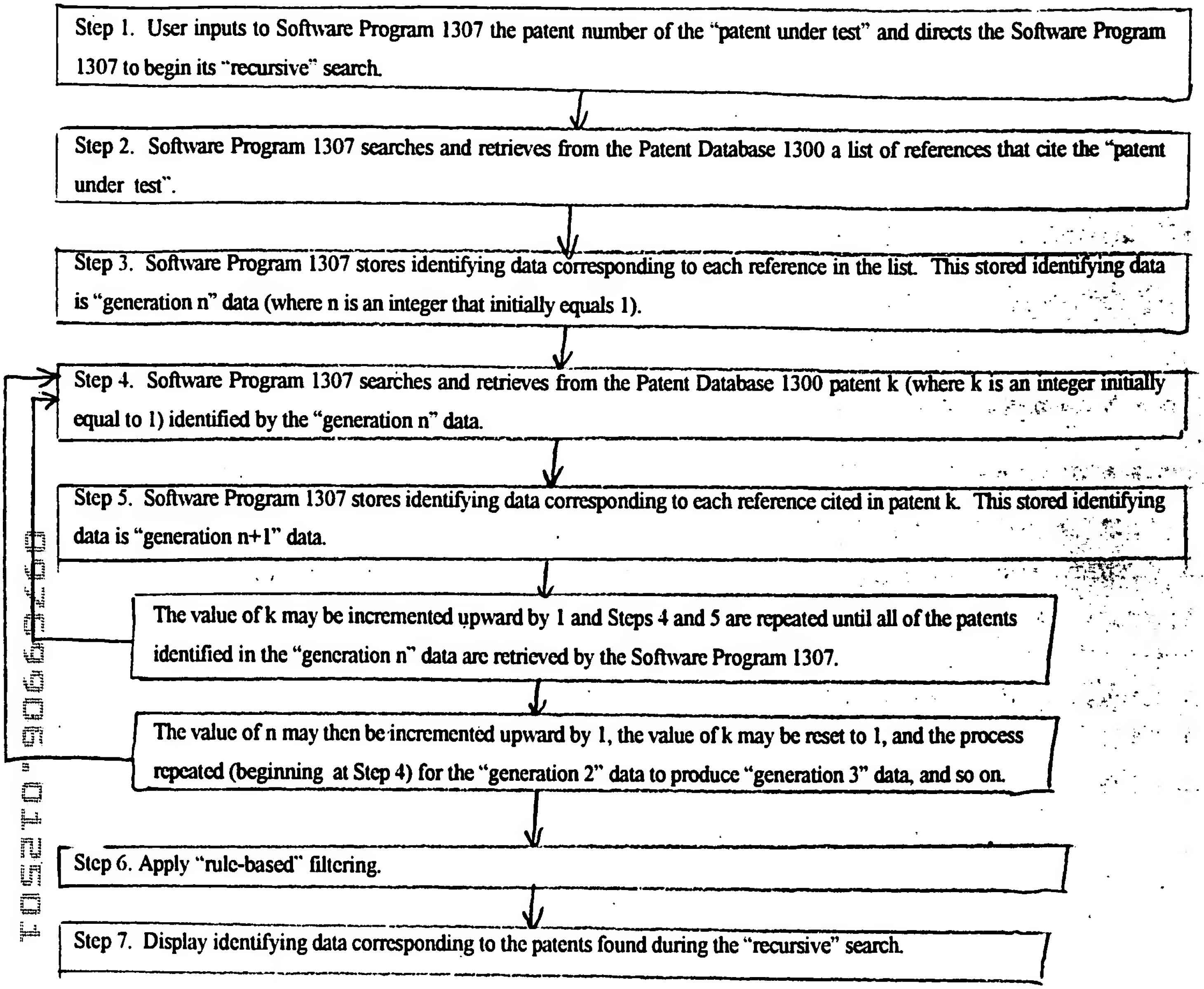

User Computer
Patent Database
1300
Software Program
1307
1301

Fig. 14



Step 1. User inputs to Software Program 1307 the patent number of the "patent under test" and directs the Software Program 1307 to begin its "recursive" search.

Step 2. Software Program 1307 searches and retrieves from the Patent Database 1300 a list of references that cite the "patent under test".

Step 3. Software Program 1307 stores identifying data corresponding to each reference in the list. This stored identifying data is "generation n" data (where n is an integer that initially equals 1).

Step 4. Software Program 1307 searches and retrieves from the Patent Database 1300 patent k (where k is an integer initially equal to 1) identified by the "generation n" data.

Step 5. Software Program 1307 stores identifying data corresponding to each reference cited in patent k. This stored identifying data is "generation n+1" data.

The value of k may be incremented upward by 1 and Steps 4 and 5 are repeated until all of the patents identified in the "generation n" data are retrieved by the Software Program 1307.

The value of n may then be incremented upward by 1, the value of k may be reset to 1, and the process repeated (beginning at Step 4) for the "generation 2" data to produce "generation 3" data, and so on.

Step 6. Apply "rule-based" filtering.

Step 7. Display identifying data corresponding to the patents found during the "recursive" search.